REPOSITIONING AND DISPLAYING AN OBJECT IN A MULTIPLE MONITOR ENVIRONMENT

RELATED APPLICATION

This application is a continuation of U.S. patent application Ser. No. 08/786,969, entitled "Robust Display Management in a Multiple Monitor Environment," filed Jan. 27, 1997, now U.S. Pat. No. 6,018,340 which is incorporated herein by reference.

BACKGROUND OF THE INVENTION

1. The Field of the Invention

This invention relates to managing cursor and window displays in a multiple monitor environment.

2. The Prior State of the Art

A typical computer system as shown in FIG. **1** includes a computer **300** having a central processing unit **304**, an input/output unit **306** and memory **302** containing various programs used by the computer **300** such as an operating system **303** and one or more application programs **305**. An end-user of the computer system communicates with computer **300** by means of various input devices (keyboard **320**, mouse **310**) which transfer information to the computer via input/output unit **306**. The computer **300** replies to this input data, among other ways, by providing responsive output to the end-user, for example, by displaying appropriate text and images on the screen of a display monitor **330**.

Operating systems often include a graphical user interface ("GUI") by which the operating systems and any applications it may be running (e.g., a word-processing program) may communicate with an end-user. A commonly used GUI implementation employs a desktop metaphor in which the screen of the monitor is regarded as a virtual desktop. The desktop is an essentially two-dimensional working template area supporting various graphical objects, including one or more display regions. As shown in FIG. **2**, information is displayed on the desktop **21** within display regions **23** (e.g., window, dialog box, pop-up menu, pull-down menu, drop-down list, icon), which are rectangular in shape. Each display region **23** may be dedicated to a specific application or to the operating system under which the applications are running. By manipulating a cursor **25** (such as with standard point & click and drag & drop techniques), an end-user can manage the display regions **23** as desired, for example, by creating new display regions or eliminating old ones, or by resizing or repositioning the display regions to fit the end-user's needs. The end-user may "activate" a particular display region and its associated application, for example, by "clicking" the cursor **25** when it appears within the desired region.

In a computer system using a single monitor **330** as shown in FIG. **1**, a problem of screen clutter may occur when an end-user has a large number of display regions open on the monitor at the same time. Screen clutter tends to confuse the end-user and reduce his or her efficiency. Moreover, end-users of certain applications (desktop publishing, CAD/CAM/CAE, video conferencing, etc.) typically will want to be able to view and use two large display regions (e.g., an editing window and an output window) at substantially the same time, but often the most useful sizes of the two windows are too large to fit side-by-side on a single monitor.

To alleviate this problem, a computer system such as that shown in FIG. **3** having two monitors **330** and **332** has been used. In the multiple monitor system of FIG. **3**, the combination of the monitor spaces (two in the case shown—one monitor space **41** corresponding to monitor **330** and a second monitor space **43** corresponding to monitor **332**) may be treated as a single, contiguous virtual desktop **45** as shown in FIG. **4**. Through appropriate cursor manipulations, an end-user may move objects, such windows A, B, C, D and cursor **25**, back and forth between the two monitor spaces **41** and **43** or may even position one of these objects (e.g., window C in FIG. **4**) so that it spans the two monitor spaces.

SUMMARY OF THE INVENTION

In one aspect of the invention, a computer having a display engine displays a window across multiple monitor spaces by determining that a display window that appears entirely within a first monitor space is to be moved or resized so that a first portion of the display window remains in the first monitor space and a second portion of the display window appears in a second monitor space. The first portion of the display window is displayed in the first monitor space by performing a bit block transfer operation, and the second portion of the display window in the second monitor space by passing the second portion of the display window through the display engine. If, however, the second monitor space has substantially the same color characteristics as the first monitor, both portions of the display window are displayed by performing a bit block transfer operation.

Windows that are moved to span monitor boundaries are re-drawn in an intelligent and efficient manner. The portion of the window that remains in the same monitor space is redrawn by moving a block of bits from one portion to another of the frame buffer for that monitor. Only the portion of the window that appears on a different monitor is "repainted" from scratch. This reduces the number of necessary pixel calculations while maintaining the color characteristics of the various monitors being used.

In another aspect of the invention, a cursor image is displayed in one of multiple available monitor spaces upon an event being generated in response to movement of an input device, for example, a mouse. While the event is pending, a new position for the cursor image is determined based on an old position of the cursor image and the movement of the input device. If the new position of the cursor image is on a different one of the monitor spaces than the old cursor image, then the cursor image is displayed at its new position if a process running in the computer is not relying on the old position of the cursor image. Otherwise, an operation is placed in a queue to display the cursor image at its new position after the event has lapsed. Subsequently, the cursor image is displayed at its new position based on a queue of pending cursor image moves. In any event, all pending cursor image moves are performed before the applications are notified of the moves.

In one aspect of the invention, objects (e.g. a cursor image, a window or other display region) are displayed in one of multiple monitor spaces by comparing a position of the object with a position of each of the monitor spaces. A monitor space is chosen from among the multiple monitor spaces based on a result of the comparison, and the object is displayed in the chosen monitor space. If the object is a cursor image, for example, the comparison may be based on a squared distance between the position of the cursor and the positions of each of the monitor spaces. The cursor image may be displayed in the monitor space (for example, at a point on the edge of the monitor space) that has the smallest squared distance between it and the cursor.

Alternatively, the comparison may be accomplished by establishing a bounding rectangle for the monitor spaces by